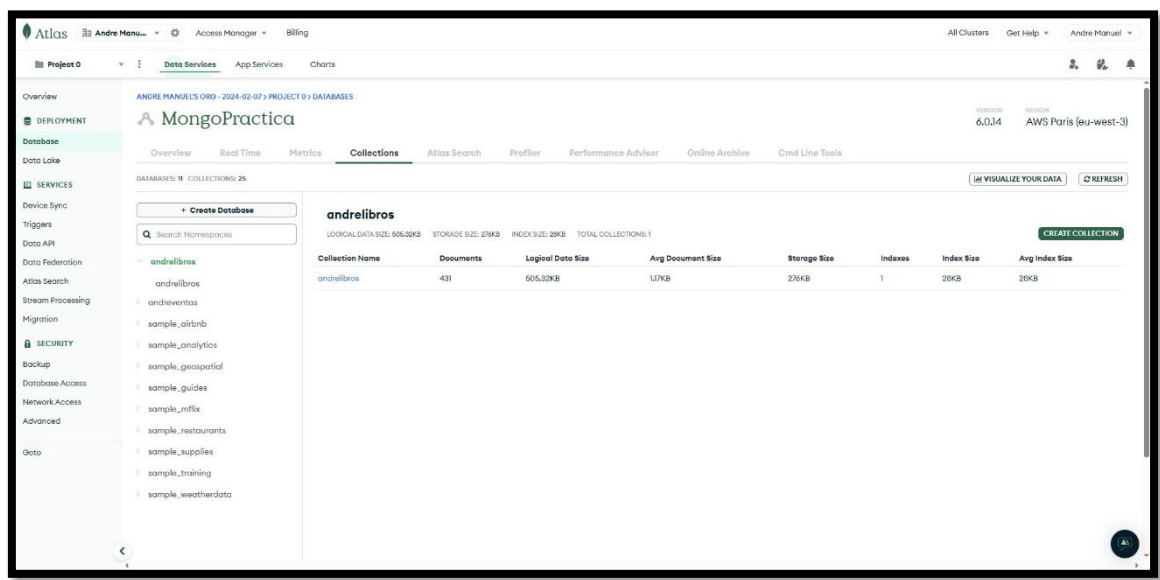


André M. Santamaría Regal

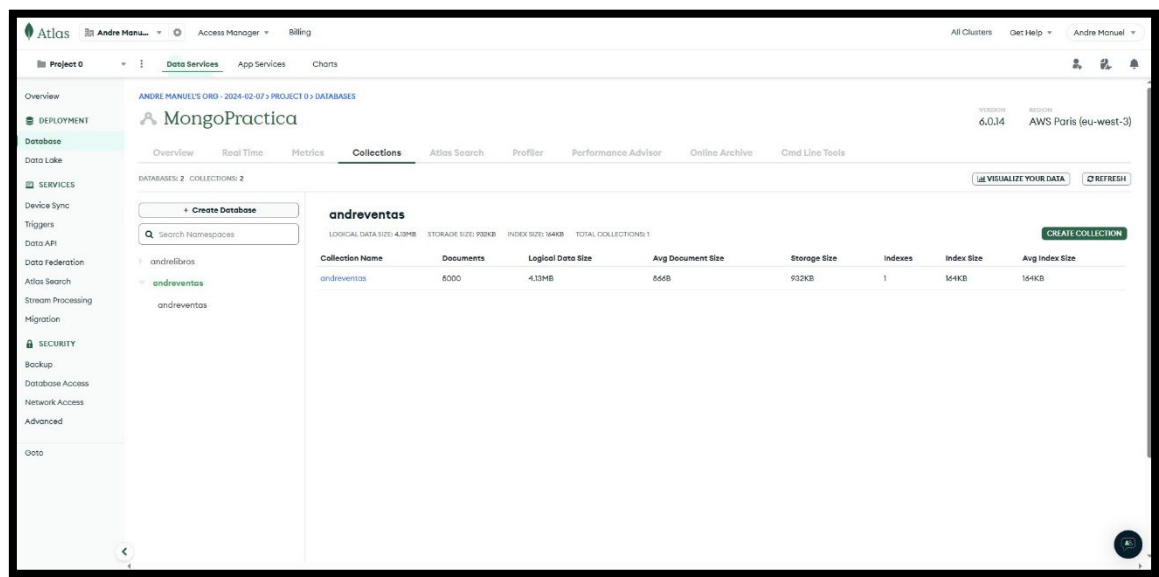
DNI: 36117001D

Enunciado

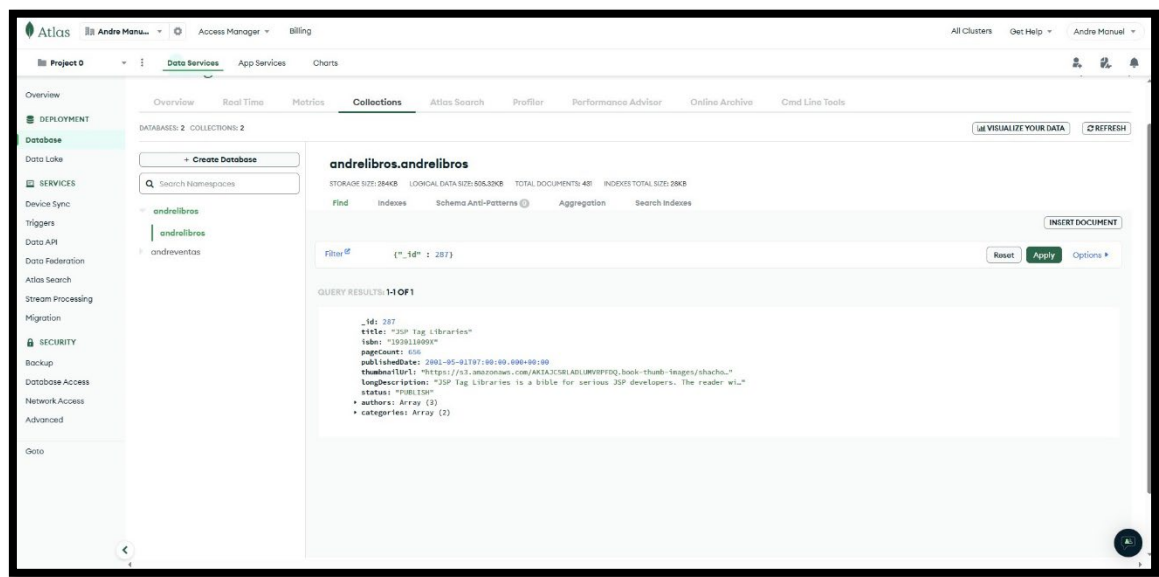
1.- Importa en una colección llamada “<tunombre>libros” el contenido del archivo books.json



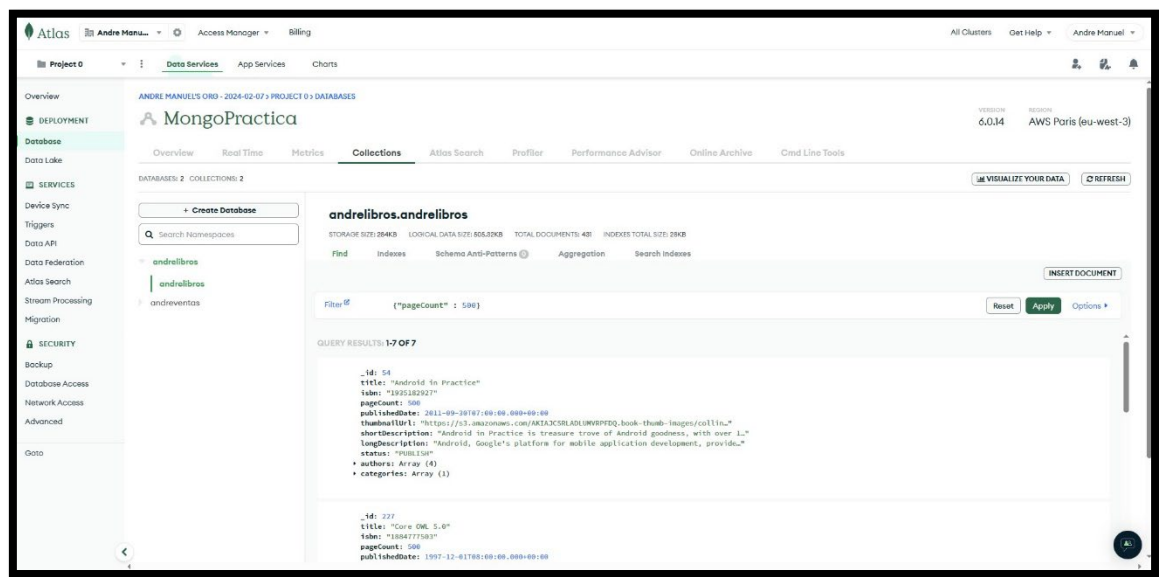
2.- Importa en una colección llamada “<tunombre>ventas” el contenido del archivo sales.json



3.- Usando la colección libros, escribe la consulta que necesitas para obtener el documento con `_id` igual a 287.



4.- Usando la colección libros, escribe una consulta que devuelva los libros con exactamente 500 páginas



5.- Usando la colección libros, escribe la consulta que devuelva los libros escritos por “Robi Sen” o por “Chris King”

The screenshot shows the Atlas Data Explorer interface for a MongoDB database named 'andrellibros'. The left sidebar contains navigation options like Overview, Databases, Collections, and Data Lake. The main panel shows the 'libros' collection with a search bar and a 'Create Database' button. Below the search bar, a JSON document is displayed for a book by Robi Sen. The document includes fields like 'title', 'author', 'year', 'isbn', 'description', and 'category'. The right sidebar shows the 'QUERY RESULTS' section with a table of 3 results. The table has columns for '_id', 'title', 'author', 'year', 'isbn', 'description', and 'category'. The results show three books by Robi Sen and Chris King.

Database: andrellibros

Collection: libros

Schema: andrellibros

Aggregation: Search Indexes

INJECT DOCUMENT

Reset Apply Options

QUERY RESULTS: 3 OF 3

_id	title	author	year	isbn	description	category
1	Android for Action, Second Edition	Robi Sen	2011	9781449126509	Android for Action, Second Edition is a comprehensive tutorial for Android developers. When it comes to mobile apps, Android can do almost anything - and will.	Android
2	Unlocking Android	Robi Sen	2011	9781449126509	Unlocking Android: A Developer's Guide provides context, hands-on tests, and more information about the Android platform based on the latest news.	Android
3	Unlocking Android	Chris King	2011	9781449126509	Unlocking Android: A Developer's Guide provides context, hands-on tests, and more information about the Android platform based on the latest news.	Android

6.- Explica con tus palabras qué hace la siguiente consulta:

`db.javilibros.find ({ "_id" : { $gt : 10 }, "_id" : { $lt : 15 } })`

La consulta mostrada en este enunciado creo que tiene la intención de que se mostraran los datos de los libros cuyo “_id” se encontrara entre el “_id” 10 y el “_id” 15, estos dos no incluidos, pero no esta bien formada por lo que el resultado que da, por como actúa mongodb por defecto al no usar ningún operador, es todos los “_id” menores a 15.

7.- Usando la colección libros, escribe la consulta que devuelva los libros publicados en enero del 2.011

The screenshot shows the MongoDB Atlas web interface. On the left, a sidebar contains navigation links: Overview, DEPLOYMENT, Database, Data Lake, SERVICES, Device Sync, Triggers, Data API, Data Federation, Atlas Search, Stream Processing, Migration, SECURITY, Backup, Database Access, Network Access, Advanced, and Oato. The 'Database' section is selected, and a search bar for 'Search Namespaces' is visible. Below it, a list of namespaces includes 'andrelibros', 'andrelibros', 'andreventos', and 'andreventos'. The 'andrelibros' namespace is selected, and its details are shown on the right. The details include 'STORAGE SIZE: 284KB', 'LOGICAL DATA SIZE: 855.3KB', 'TOTAL DOCUMENTS: 438', and 'INDEXES TOTAL SIZE: 28KB'. A 'Find' button is present, and a filter bar shows the query: `{ 'publishedDate' : { '$gte' : ISODate('2011-01-01'), '$lt' : ISODate('2011-02-01') } }`. The query results are displayed as a list of documents. The first document is:

```
{
  "_id": 2,
  "title": "Android in Action, Second Edition",
  "isbn": "1395182722",
  "pageCount": 592,
  "publishedDate": "2011-01-14T00:00:00.000-00:00",
  "thumbnailUrl": "https://s3.amazonaws.com/AMIAJCSRLADLUMVRFDDQ-book-thumb-images/ables...",
  "shortDescription": "Android in Action, Second Edition is a comprehensive tutorial for And...",
  "longDescription": "When it comes to mobile apps, Android can do almost anything... and wit...",
  "status": "PUBLISHED",
  "authors": Array (2),
  "categories": Array (1)
}
```

 The second document is:

```
{
  "_id": 155,
  "title": "Camel in Action",
  "isbn": "1395182760",
  "pageCount": 375,
  "publishedDate": "2011-01-04T00:00:00.000-00:00",
  "thumbnailUrl": "https://s3.amazonaws.com/AMIAJCSRLADLUMVRFDDQ-book-thumb-images/ibsen...",
  "shortDescription": "Camel in Action is for developers working with integration of any kind...",
  "longDescription": "Apache Camel is a Java-based toolkit that makes it easy to implement s...",
  "status": "PUBLISHED",
  "authors": Array (3)
}
```

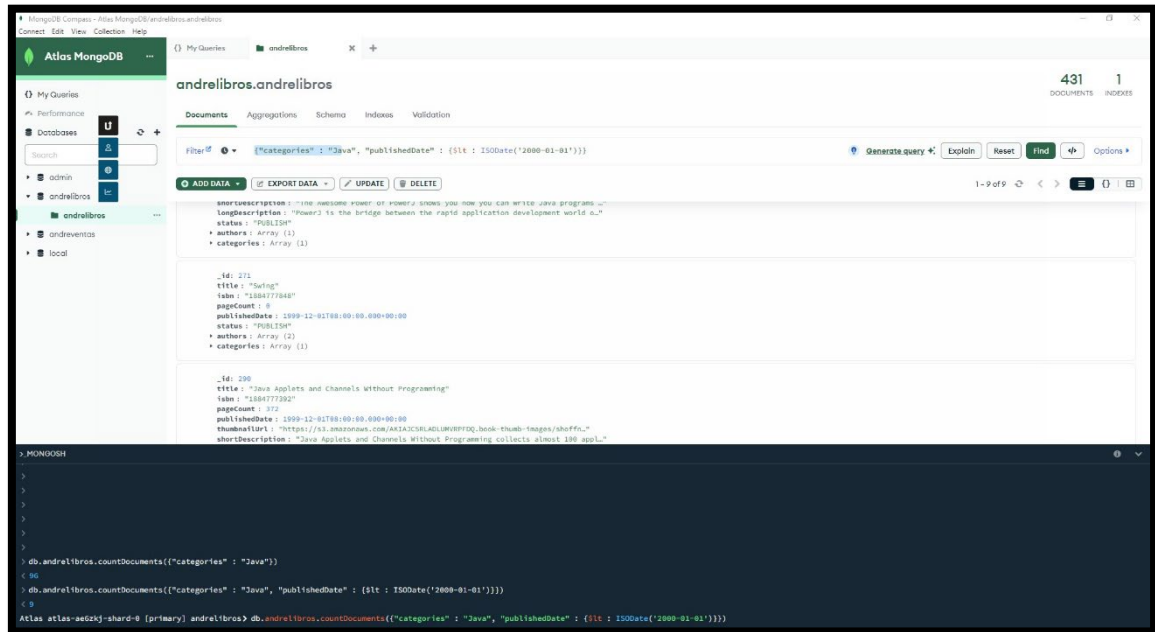
 At the bottom, the system status is 'All Good' and the version is '6.0.204 MongoDB, Inc.'. The footer includes links for Status, Terms, Privacy, Atlas Blog, and Contact Sales.

8.- Usando la colección libros, escribe la consulta que devuelve todos los libros en cuyo título podemos encontrar la palabra “Microsoft”

The screenshot shows the MongoDB Atlas web interface. On the left is a navigation sidebar with sections like Overview, DEPLOYMENT, Database, SERVICES, SECURITY, and Goto. The 'Database' section is active, showing a list of databases: andrellibros, andrellibros, andreventas, and andreventas. The main panel displays the 'andrellibros.andrellibros' collection. At the top, it shows statistics: STORAGE SIZE: 264KB, LOCAL DATA SIZE: 856.52KB, TOTAL DOCUMENTS: 431, INDEXES TOTAL SIZE: 29KB. Below this are tabs for Find, Indexes, Schema And Patterns, Aggregation, and Search Indexes. The 'Find' tab is selected, showing a filter bar with the query `{"title" : /Microsoft/}`. Below the filter bar, it says 'QUERY RESULTS: 1-3 OF 3'. Three document snippets are visible, each with fields like `_id`, `title`, `isbn`, `pageCount`, `publishedDate`, `thumbnailUrl`, `shortDescription`, `longDescription`, `status`, `authors`, and `categories`. The first document has a title 'Microsoft Office Essentials'. The second document has a title 'Microsoft.NET for Programmers'. The third document has a title 'Microsoft.NET for Programmers'. At the bottom of the interface, there is a footer with 'System Status: All Good' and copyright information for MongoDB, Inc.

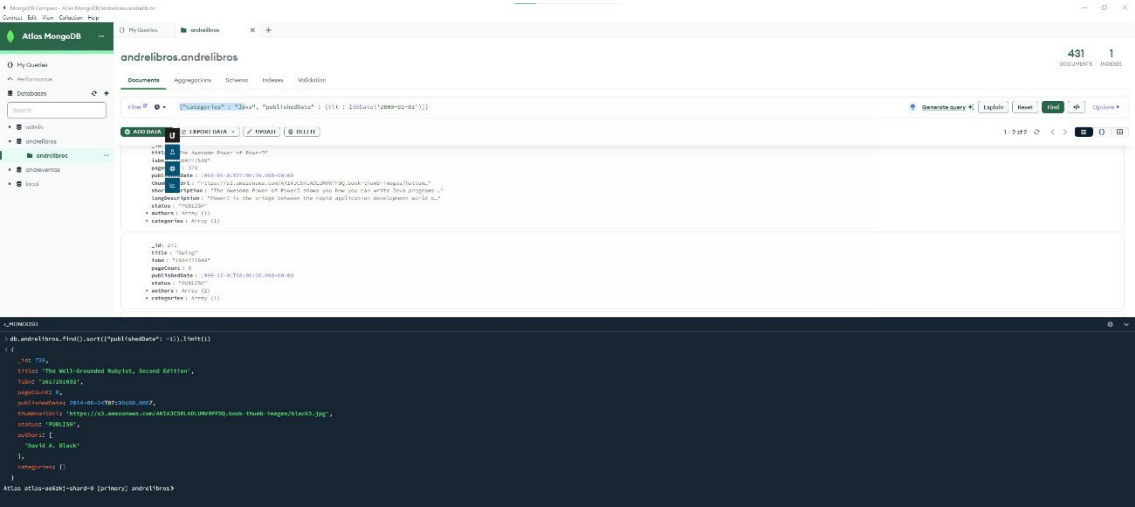
Tambien se puede utilizar la consulta `{"title" : {$regex : "Microsoft"}}`

9.- Usando la colección libros, escribe una consulta que indique cuantos libros en la categoría de “Java” se han publicado antes del año 2.000.



He utilizado el método `countDocuments()` en vez del `count()` por el mensaje que da mongodb sobre el método `count()`, ya que dice que ese método esta deprecated desde la versión 4.0 de mongoDB.

10.- Usando la colección libros, escribe una consulta que indique solamente el título y la fecha de publicación del libro más recientemente publicado.



11.- En la colección ventas, explica dónde está el problema en la segunda consulta de agregados de esta captura

```
applies> use supplies
switched to db supplies
supplies> db.sales.aggregate([ {$group: {"_id":"$storeLocation"}}])
[
  {
    _id: 'Austin' },
  {
    _id: 'London' },
  {
    _id: 'Seattle' },
  {
    _id: 'Denver' },
  {
    _id: 'San Diego' },
  {
    _id: 'New York' }
]
supplies> db.sales.aggregate([ {$group: {"_id":"$storeLocation"}}, {$sort:{"storeLocation":1}}])
[
  {
    _id: 'Austin' },
  {
    _id: 'London' },
  {
    _id: 'Seattle' },
  {
    _id: 'Denver' },
  {
    _id: 'San Diego' },
  {
    _id: 'New York' }
]
supplies> _
```

La segunda consulta de agregación falla porque después del operador \$group, el campo por el que intentas ordenar (storeLocation) no existe en los documentos resultantes. Deberías ordenar por _id en lugar de storeLocation, ya que _id es el campo que contiene la ubicación de la tienda después de la agrupación.

12.- En la colección de ventas, indica la consulta que usarías para obtener un listado en el que aparezca cuantas unidades de cada elemento se han vendido en cada tienda y que aparezca ordenado alfabéticamente por tienda y descendientemente por número de elementos vendidos. Debería quedar como en esta captura.

```

tienda: 'Austin', item: 'envelopes', unidades: 3490 },
tienda: 'Austin', item: 'binder', unidades: 3331 },
tienda: 'Austin', item: 'notepad', unidades: 3128 },
tienda: 'Austin', item: 'pens', unidades: 1821 },
tienda: 'Austin', item: 'printer paper', unidades: 1597 },
tienda: 'Austin', item: 'laptop', unidades: 1035 },
tienda: 'Austin', item: 'backpack', unidades: 949 },
tienda: 'Denver', item: 'binder', unidades: 7986 },
tienda: 'Denver', item: 'envelopes', unidades: 7832 },
tienda: 'Denver', item: 'notepad', unidades: 6276 },
tienda: 'Denver', item: 'pens', unidades: 4188 },
tienda: 'Denver', item: 'printer paper', unidades: 3727 },
tienda: 'Denver', item: 'backpack', unidades: 2121 },
tienda: 'Denver', item: 'laptop', unidades: 2025 },
tienda: 'London', item: 'envelopes', unidades: 4037 },
tienda: 'London', item: 'binder', unidades: 3871 },
tienda: 'London', item: 'notepad', unidades: 3278 },
tienda: 'London', item: 'pens', unidades: 2232 },
tienda: 'London', item: 'printer paper', unidades: 2042 },
tienda: 'London', item: 'backpack', unidades: 1177 }

```

La consulta que he realizado para sacar una imagen similar a la captura es:

```

db.andreventas.aggregate([
  { $unwind : "$items"},
  { $group : { "_id": {"tienda" : "$storeLocation", "item" : "$items.name"}, "unidades" : { $sum :
    "$items.quantity" } }},
  { $sort : { "_id.tienda" : 1, "unidades": -1 }},
  { $project : { "_id" : 0, "tienda" : "$_id.tienda", "item" : "$_id.item", "unidades" : "$unidades" } }
])

```

```

MongoDB
tienda: 'Austin',
unidades: 3490
}
{
  tienda: 'London',
  item: 'pens',
  unidades: 2232
}
{
  tienda: 'London',
  item: 'printer paper',
  unidades: 2042
}
{
  tienda: 'London',
  item: 'backpack',
  unidades: 1177
}
Type "it" for more

```

13.- Indica la consulta que muestre las 3 tiendas que más han facturado donde se vea su nombre y la cantidad facturada. Será necesaria la siguiente expresión para calcular el precio por cantidad. {\$multiply: [“\$campo1”, “\$campo2”] }

```
db.andreventas.aggregate([
  { $unwind: "$items"},
  { $project: {storeLocation: 1, facturacion: { $multiply: ["$items.price", "$items.quantity"]}}} ,
  { $group: { "_id": {"tienda": "$storeLocation"}, "total": { $sum: "$facturacion" } }},
  { $sort: {total: -1}},
  { $limit: 3},
  { $project: { "_id": 0, "tienda": "$_id.tienda", "total facturado": "$total" }}}]
```